Automated multi-layer optical design via deep reinforcement learning

IOP Publishing Mach. Learn.: Sci. Technol. 2 (2021) 025013 https://doi.org/10.1088/2632-2153/abc327

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
30 June 2020

REVISED
30 September 2020

ACCEPTED FOR PUBLICATION
20 October 2020

PUBLISHED
3 February 2021

PAPER

Automated multi-layer optical design via deep reinforcement learning

Haozhu Wang[1], **Zeyu Zheng**[1], **Chengang Ji**[2] **and L Jay Guo**[1]

[1] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, United States of America
[2] Ningbo Inlight Technology Co., Ltd, Ningbo, Zhejiang, People's Republic of China

E-mail: hzwang@umich.edu **and** guo@umich.edu

Keywords: reinforcement learning, optical design, optimization

Abstract

Optical multi-layer thin films are widely used in optical and energy applications requiring photonic designs. Engineers often design such structures based on their physical intuition. However, solely relying on human experts can be time-consuming and may lead to sub-optimal designs, especially when the design space is large. In this work, we frame the multi-layer optical design task as a sequence generation problem. A deep sequence generation network is proposed for efficiently generating optical layer sequences. We train the deep sequence generation network with proximal policy optimization to generate multi-layer structures with desired properties. The proposed method is applied to two energy applications. Our algorithm successfully discovered high-performance designs, outperforming structures designed by human experts in task 1, and a state-of-the-art memetic algorithm in task 2.

1. Introduction

Optical multi-layer films have been widely used in many applications, such as broadband filtering [1], photovoltaics [2], radiative cooling [3], and structural colors [4]. The design of optical multi-layer films is a combinatorial optimization problem that requires one to choose the best combination of materials and layer thicknesses to form a multi-layer structure. Researchers and engineers often make such designs based on their physical intuition. However, a completely human-based design process is slow and often leads to sub-optimal designs, especially when the design space is enormous. Thus, computational methods for designing optical multi-layer structures, including evolutionary algorithms [5–7], needle optimization [8], and particle swarm optimization [9], have been proposed to tackle this problem. All of these previous methods frame the optical design task as an optimization problem and aim to synthesize a structure that meets user-specified design criteria. However, these methods for optical design are based entirely on heuristic search, i.e. they do not learn a model to solve the design problems. When the heuristic approach is sub-optimal for a task, the search process may fail to identify a high-performance design.

In contrast, deep reinforcement learning (DRL) is a learning framework that learns to solve complex tasks through an trial-and-error process. It is proven to be highly scalable for solving large-scale and complicated tasks [10, 11]. Researchers have successfully applied DRL to various combinatorial optimization problems [12–15]. Unlike heuristic-based search, reinforcement learning methods learn a model using the reward signal [16] and do not depend on hand-crafted heuristics. On some combinatorial optimization tasks, DRL has been shown to outperform classic heuristic search methods [17]. Recently, researchers applied DRL on designing optical devices with a structure template [18, 19], where the number of layers is fixed. However, when designing the optical multi-layer films, we often do not know the optimal structure template. Thus, the previous DRL approaches are not suitable for multi-layer designs. In addition to DRL, deep learning-enabled inverse design methods have seen great development in recent years [20–22]. These inverse design models learn a mapping between design targets and design parameters using a static training set, which allows users to efficiently retrieve designs that match design targets. However, if a design target does not lie within the training datasets used for training the inverse design model, we will not be able to obtain the corresponding

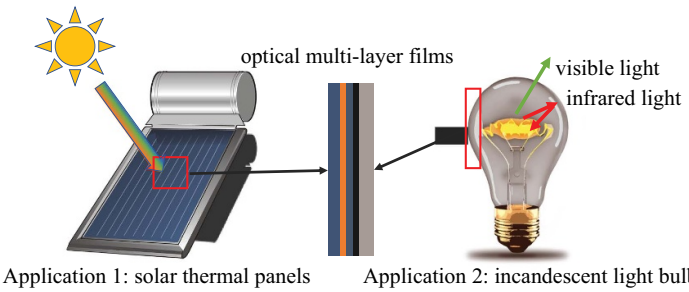

Figure 1. Two energy applications of optical multi-layer films. For solar thermal panels, we can use multi-layer films as ultra-wideband absorbers to enhance light absorption efficiency. For incandescent light bulbs, we can coat multi-layer films on them to improve luminous efficiency by reflecting infrared light while transmitting visible light.

design using the inverse design model. For our performance optimization task, the optimal design is often not covered by a static training dataset. Otherwise, it would mean that the optimization task has already been solved through the training dataset collection process. Thus, reinforcement learning is more suitable than deep-learning-based inverse design methods when users want to optimize the design performance.

Because the multi-layer optical design task is equivalent to a sequence generation problem, we propose a DRL method called optical multi-layer proximal policy optimization (OML-PPO) that can generate near-optimal multi-layer structures. The proposed method uses a state-of-the-art DRL algorithm PPO to train a deep recurrent neural network that outputs near-optimal optical designs. We introduce two novel designs for the deep recurrent neural network to allow it to efficiently explore the design space. With an ablation study, we show that the proposed neural network architecture enables the RL agent to explore the design space efficiently.

We applied the proposed method to two optical design tasks that are relevant to energy applications (figure 1): (1) ultra-wideband absorbers that can enhance light-harvesting efficiency, e.g. for thermal photovoltaics and photothermal energy conversion; and (2) incandescent light bulb filters that can improve light bulb efficiency in emitting visible light. On the task of designing ultra-wideband absorbers, we show that OML-PPO can reliably discover high-performance designs. A 5-layer structure with 97.64% average absorption over the wavelength range (400, 2000) nm is discovered by OML-PPO, outperforming a previously reported structure using the same number of layers with 95.37% average absorption. When applied to generate absorbers with more layers, OML-PPO discovers a 14-layer structure that achieves near-perfect 99.24% average absorption. We also applied our method to design a 42-layer incandescent light bulb filter and achieved an enhancement factor of 16.60, which is 8.5% higher than a 41-layer structure designed by a state-of-the-art memetic algorithm. Our results demonstrate that the proposed algorithm is efficient at discovering near-optimal designs and is scalable to complicated design problems. We summarize our contributions:

(a) We frame the multi-layer optical design task as a sequence generation problem and develop a DRL method (OML-PPO) for solving this task.
(b) We propose a novel deep sequence generation network that allows efficient exploration of the optical design space.
(c) On two optical design tasks, we demonstrate that our method is effective in discovering near-optimal solutions for complicated design tasks.

2. Related work

Researchers have developed reinforcement learning methods for solving various combinatorial optimization problems. In [12], the authors trained a pointer network [23] to solve the traveling salesman problem (TSP). Khalil *et al* [13] combined graph embedding and RL for solving a diverse set of combinatorial optimization problems including the minimum vertex cover, maximum cut, and TSP. Chen and Tian [24] proposed a method to learn policies that can rewrite the heuristics in existing solvers for combinatorial optimization problems. Lu *et al* [17] showed that RL-based method could outperform a classic operation research algorithm in terms of both average cost and time efficiency.

Many real-life applications can be formalized as sequence generation problems [15, 25–27]. In [25], the authors integrated RL and seq2seq to automatically generate a response by simulating the dialogue between

two agents. In [27], the authors proposed a model-based variant of PPO to deal with the large-batch, low round setting for biological sequence design [27]. Mirhoseini *et al* [15] combined graph neural networks with RL for sequentially placing devices on a chip. These previous works all trained sequence generation models using policy gradient algorithms. In this work, we introduced a sequence generation network architecture tailored to the optical design task. Additionally, we combined local search with DRL for finetuning the thicknesses of the generated layers.

Deep-learning-based inverse design [20–22] has been gaining popularity in recent years. In [20], the authors trained convolutional neural networks to directly predict design parameters using the design target as the input to the network. Liu *et al* [22] trained a generative adversarial network (GAN) to inversely design optical devices by generating 2D shapes of the optical structure. However, these approaches all rely on a curated training set that contains diverse examples. When our goal is to push the performance limit of certain devices, the near-optimal structures is unlikely to be within the training data distribution. Thus, these static methods are not appropriate for optimizing design performances. Our proposed method tackles this problem by actively searching the design space to generate high-performance designs via reinforcement learning. In [28], the authors also developed an active search process by adding additional high-quality data to augment the initial training set. However, their approach requires the users to retrain the neural network with the augmented dataset while our RL-based method accomplishes the design task within one training process.

3. Methods

Multi-layer films can be treated as sequences. Each layer is represented as $s_l = (m_l, d_l)$. We can represent such a structure with N layers as $\mathcal{S} = \{(m_0, d_0), (m_1, d_1), (m_2, d_2), \ldots, (m_{N-1}, d_{N-1})\}$, where m_l and d_l denote the material and the thickness of the lth layer (counting from the top), respectively. When designing optical multi-layer films, we hope to synthesize a sequence that has the desired target spectral response $\tilde{T}$. Thus, the design task is equivalent to a sequence generation problem, where we generate m and d in each step. Generation tasks such as dialogue generation [25], molecule generation [26], and biological sequence generation [27] have been widely studied by machine learning researchers. In these works, researchers train a neural network as a generator for synthesizing sequences. Because we do not have ground-truth data for optimal design tasks, we apply reinforcement learning [16] to train the sequence generator.

3.1. Sequence generation network

To generate the optical layer sequences, we use a recurrent neural network (RNN) [29]. Unlike simple feed-forward neural networks, RNNs maintain a hidden state h that contains useful information from the history of the sequence. Thus, RNNs are suitable for tasks that require memorizing history and have been widely used in sequence generation tasks [30]. Gated recurrent units (GRUs) [31] and long short-term memory networks (LSTMs) [29] are two popular variants of RNNs. Researchers have previously found that the empirical performance of GRUs and LSTMs is similar. Because GRUs have a simpler structure than LSTMs and require fewer parameters to train, we choose to use a GRU for generating the optical multi-layer structures. Similar to sampling words from a dictionary when generating a sentence, we sample the material m_l from a fixed set of materials $\mathcal{M}$ for each layer. Though the thickness d_l is intrinsically a continuous variable, we choose to sample the thickness from a set of discrete values $\mathcal{D}$ to reduce the size of the exploration space. Later, we apply quasi-Newton methods [32] to finetune the layer thicknesses of the generated structure for further performance improvement.

Our optical multi-layer sequence generation network consists of a GRU and two multi-layer perceptrons (MLPs) [33]. At generation step l, the GRU takes its own output from the previous step $s_{l-1} = (m_{l-1}, d_{l-1})$ and the previous hidden state h_l as the inputs to compute the hidden state h_l. This auto-regressive generation process allows the GRU to remember what has been generated so far. To generate the material and thickness for layer l, the hidden state h_l of the GRU is inputted to two MLPs. One of the MLPs outputs logits vector $\sigma_{m_l} \in \mathbb{R}^{|\mathcal{M}|+1}$ corresponding to all possible materials and an end-of-sequence token (EOS). The other MLP outputs a thickness logits vector $\sigma_{d_l} \in \mathbb{R}^{|\mathcal{D}|}$ corresponding to all allowable thicknesses in the set $\mathcal{D}$. Then, we transform these logits vectors with the *softmax* function to obtain proper probability distributions. Finally, the material and thickness are sampled from their corresponding distributions. The generation process will stop either when the length reaches the maximum length L set by the user or when the EOS token is sampled. Thus, the number of layers N of a generated structure is always lower than or equal to the maximum sequence length L. The process for generating a sequence is illustrated in figure 2.

3.1.1. Non-repetitive gating

The aforementioned material sampling procedure does not prevent the situation where the same material is sampled for adjacent layers. However, such consecutive layers of the same material are equivalent to a single

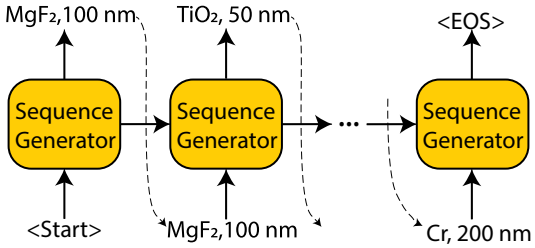

Figure 2. Optical multi-layer design as sequence generation. The generation process will stop when either the EOS token is sampled, or the length of the sequence reaches the maximum allowed length L.

thicker layer. Thus, allowing the sequence generator to generate the same material for adjacent layers leads to redundant computation. Moreover, doing so increases the exploration space size and makes the search problem harder. Thus, we introduce a non-repetitive gating function that removes the logit element corresponding to the most recently sampled material to prevent the sequence generator from generating the same materials in a row. This gating function is a matrix $I_{NR} \in \mathbb{R}^{|\mathcal{M}| \times (|\mathcal{M}|+1)}$ formed by removing the row corresponding to the most recently sampled material from an identity matrix. When multiplied with the logits vector σ_{m_l}, the element corresponding to that material will be removed, i.e. $\sigma'_{m_l} = I_{NR} \cdot \sigma_{m_l} \in \mathbb{R}^{|\mathcal{M}|}$. Then, we pass the transformed logit vector σ'_{m_l} to the softmax layer to obtain the sampling probability. By doing so, we set the sampling probability for the recurring material to 0. With the non-repetitive gating, the generated material sequence is guaranteed to have different materials for adjacent layers. Note that, we do not apply the gating function for the first generation step because there is no previously sampled material.

3.1.2. Auto-regressive generation of material and thickness

Because the proper thickness of a layer should depend on the material, we input the sampled material m_l to the thickness MLP in addition to the hidden state h_l. A similar approach has been applied in RL problems where the actions are dependent on each other [11]. Instead of using a one-hot vector to represent the material, we train a material embedding matrix $emb \in \mathbb{R}^{|\mathcal{M}| \times d}$ together with the sequence generator network. Each row $emb_m \in \mathbb{R}^d$ of the embedding matrix is a continuous representation of one material, where d is the embedding size. Using an embedding allows us to use a large number of materials without significantly increasing the dimensionality of the material representation. The material embedding vector for the sampled material emb_{m_l} is concatenated with the hidden state h_l to form the input $[emb_{m_l}, h_l]$ to the material MLP.

The full sequence generator architecture is plotted in figure 3(a). To understand the effect of non-repetitive gating and modeling the dependency between the material and the thickness, we compare the proposed OML-PPO architecture against a baseline architecture Experiment section.

3.2. Reinforcement learning training

We train the sequence generation network with reinforcement learning. The goal of reinforcement learning is to maximize expected cumulative rewards $G = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t]$ by learning a policy $\pi_\theta(a|s)$ that can map a state s to an action a. Here, γ is the discount factor that penalizes future rewards and r_t is the reward at step t. The sequence generation network described above serves as the policy.

We represent the state at the lth generation step as the concatenation of the last layer information and the GRU hidden state, i.e. $s_l = [(m_{l-1}, d_{l-1}), h_l]$. The actions a_l correspond to the material and thickness (m_l, d_l) of the current layer. We set the reward to be 0 for all generation steps except the final step. At the final step (i.e. the structure $\mathcal{S}$ has been completely generated), we compute the spectrum of the generated structure with an optical spectrum calculation package TMM [34] and assign the final reward based on how well the structure spectrum matches with the target spectrum. We also tried to calculate the spectrum following every generation step and assign intermediate rewards. However, this dense-reward approach is slow and does not lead to improved performance. Thus, we only report the final-only approach here. We set the discount factor $\gamma = 1$. Thus, the cumulative reward G for the generated sequence $\mathcal{S}$ is simply the reward at the final step, which is defined as one minus the mean absolute error between the spectrum of the generated structure and the target spectrum:

$$G(\mathcal{S}) = 1 - \frac{1}{K}\sum_{k=0}\frac{1}{J}\sum_{j=0}^{J-1}|T^{\mathcal{S}}(\lambda_j, \delta_k) - \tilde{T}(\lambda_j, \delta_k)|, \quad (1)$$

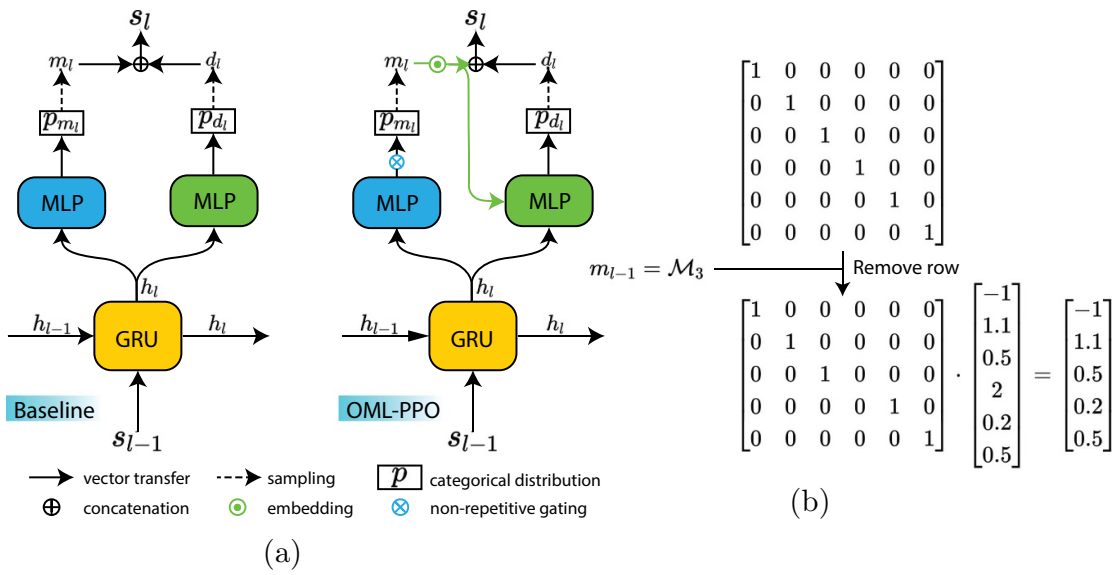

Figure 3. Neural network architectures for generating optical multi-layer films. (a) We show one generation step in the plot. The hidden state h_l of the GRU is passed to two MLPs to output material and thickness probabilities, respectively. The actual material and thickness for layer l are sampled from categorical distributions parametrized by p_{m_l} and p_{d_l}. Built-upon the baseline architecture, our proposed model adds a non-repetitive gating function and auto-regressive connection between the sampled material and the thickness MLP. (b) Illustration of how the non-repetitive gating works. Here we suppose there are a total of five materials. Thus, the gating matrix is of dimension 5×6.

where $T^{\mathcal{S}}(\lambda_j, \delta_k)$ is the spectrum of the generated structure $\mathcal{S}$ at wavelength λ_j under incidence angle δ_k. Because $T \in [0, 1]$, the cumulative reward is always non-negative. The reward value will become higher as the spectrum $T^{\mathcal{S}}$ gets closer to the target spectrum $\tilde{T}$ until it reaches 1 when the structure spectrum perfectly matches with the target spectrum.

During training, the sequence generator π_θ actively generates new structures and receive rewards. Our goal is to maximize the expected rewards for structures sampled from the sequence generation network:

$$J(\theta) = \mathbb{E}_{\mathcal{S}\sim\pi_\theta}[G(\mathcal{S})]. \tag{2}$$

Based on the calculated rewards for generated sequences, the agent adjusts its parameters θ with gradient ascent so that future rewards can be improved. Here, we use a policy gradient algorithm to compute the gradient $\nabla_\theta J(\theta)$ for updating the sequence generator π_θ. From the policy gradient theorem [16, 35], we have

$$g = \nabla_\theta J(\theta) = \mathbb{E}_{\mathcal{S}\sim\pi_\theta}\left[A(\mathcal{S})\nabla_\theta \log P_\theta(\mathcal{S})\right], \tag{3}$$

where $P_\theta(\mathcal{S}) = \prod_{l=0}^{N-1} p_\theta(m_l|s_{l-1}, h_{l-1}) \cdot p_\theta(d_l|m_l, s_{l-1}, h_{l-1})$ is the probability of sampling a structure $\mathcal{S}$ from the generator network π_θ and $A(\mathcal{S})$ is the estimated advantage function [36], which measures the performance of the generated sequence $\mathcal{S}$ compared against the average performance of structures sampled from π_θ.

Instead of directly updating the sequence generator using equation 3, we use a state-of-the-art policy gradient algorithm *proximal policy optimization* (PPO) [35] to compute the policy gradient from a surrogate objective function:

$$g = \nabla_\theta \mathbb{E}_{\mathcal{S}\sim\pi_\theta}\left[\min\left(r(\theta)A_{\theta_v}(\mathcal{S}), clip\left(r(\theta), 1-\epsilon, 1+\epsilon\right)A_{\theta_v}(\mathcal{S})\right)\right], \tag{4}$$

where $r(\theta) = \frac{P_\theta(\mathcal{S})}{P_{\theta_{\text{old}}}(\mathcal{S})}$ is the importance weight that measures the distance between the policies before and after the gradient update. The *clip* function disincentivizes large update steps to the policy, where ε is a hyperparameter that affects the actual update size. Here, the advantage A_{θ_v} is estimated by *generalized advantage estimation* (GAE) [36], which achieves a good balance between bias and variance of the estimated gradients. θ_v is the model parameters for a critic network that is trained together with the sequence generator. Compared to the vanilla policy gradient and actor-critic algorithms, PPO is more sample-efficient because it allows multi-step updates using the same batch of trajectories. Previous results show that PPO can achieve state-of-the-art performance on many tasks [35]. With the computed policy gradient, the sequence

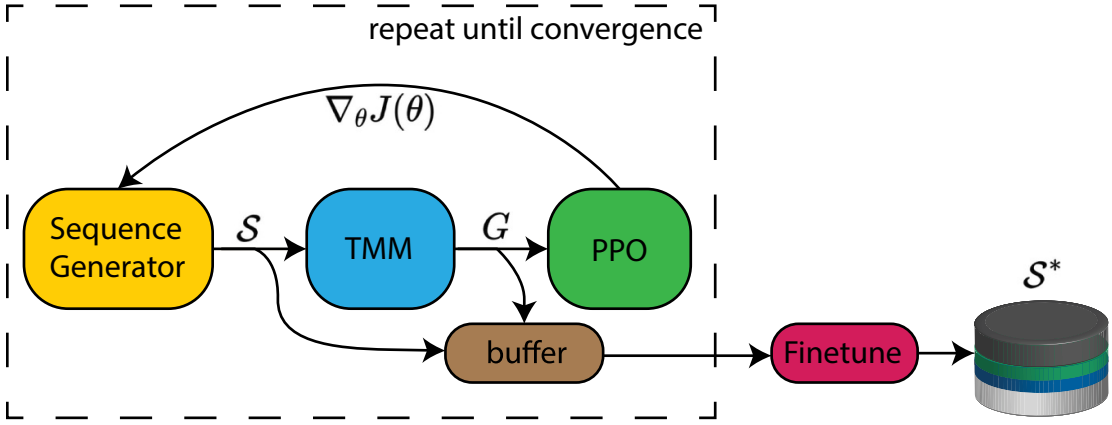

Figure 4. Pipeline of the sequence generator training process. We first generate multi-layer structures using the sequence generator π_θ. The spectrum of the generated structures are simulated by the TMM module. Next, PPO algorithm is applied to compute the policy gradient g for updating the sequence generator model. We keep pushing the best discovered structure into a buffer with size 1. This process is repeated until convergence. Finally, we finetune the layer thicknesses to obtain the design.

generator model parameters are updated using the Adam optimizer [37]. The model training process is summarized in figure 4. Similar to the *active search* approach in Bello *et al* [12], we output the best structure discovered throughout the entire training process as the final design. The pseudocode that summarizes our design generation process is given in algorithm 1.

Our model is implemented using PyTorch [38] and Spinning Up [39]. The data used in this study and our code are publicly available[3].

Algorithm 1: OML-PPO.

Input: target $\tilde{T}$, number of epochs K, batch size B, maximum length L
Output: Optical multi-layer sequence $\mathcal{S}^*$

Initialize sequence generator parameters θ
Initialize critic network parameters θ_v
Initialize best design $\mathcal{S}^*$
for k = 1, …, K **do**
  $\mathcal{S}_i \sim \texttt{SampleDesign}(L, B, \theta)$
  $\mathcal{S}^* \leftarrow \texttt{SelectBest}(\{\mathcal{S}_i\}, \mathcal{S}^*, \tilde{T})$
  $\theta, \theta_v \leftarrow \texttt{PPOUpdate}(\{\mathcal{S}_i\}, \theta, \theta_v)$
end
$\mathcal{S}^* \leftarrow \texttt{QuasiNewton}(\mathcal{S}^*, \tilde{T})$

4. Experiment

We applied the proposed method to two optical design tasks that are relevant to energy applications, i.e. (1) designing ultra-wideband absorbers and (2) designing incandescent light bulb filters. The designed ultra-wideband absorbers can help solar thermal panels to absorb the sunlight more efficiently and the light bulb filter can enhance incandescent light bulb efficiency in emitting visible light while suppressing the radiation in the infrared range that represents energy loss. We also did an ablation study to understand the effect of non-repetitive gating and auto-regressive materials/thickness sampling.

Performance evaluation: In task 1 ultra-wideband absorber design, we measure the quality of the designed structure by *average absorption*. In task 2 incandescent light bulb filter, we calculate the visible light *enhancement factor* to measure the performance of designed structures.

4.1. Task 1: ultra-wideband absorber

Firstly, we apply our algorithm to the task of designing an ultra-wideband absorber for the wavelength range (400, 2000) nm. We choose the target spectrum as a constant 100% absorption under normal light incidence angle (i.e. the light is shining at the absorber at a right angle) to represent an ideal broadband absorber. This task has been previously studied by Yang *et al* [1] based on physical models, where the broadband absorption

[3] https://github.com/hammer-wang/oml-ppo.

Table 1. Available materials for constructing the ultra-wideband absorber.

Ag	Al	Al_2O_3	Cr	Fe_2O_3	Ge	HfO_2	MgF_2
Ni	Si	SiO_2	Ti	TiO_2	ZnO	ZnS	ZnSe

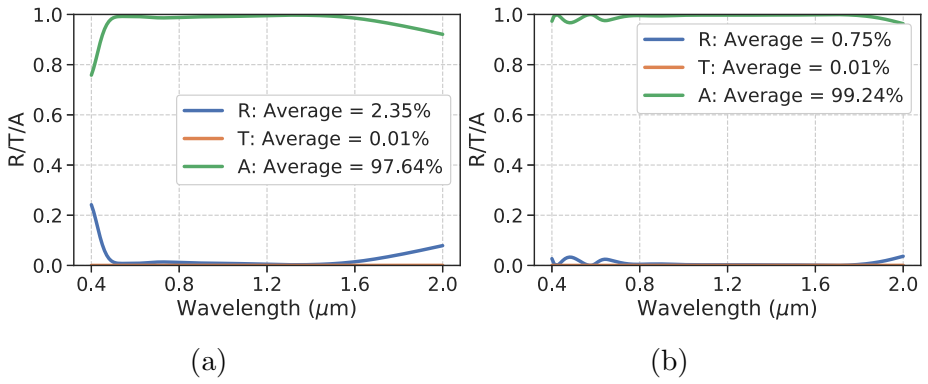

Figure 5. Normal incidence spectrum for the best discovered absorber structures with 5 and 14 layers. R: reflection, T: transmission, A: absorption. We design the multi-layer thin film to have high *absorption* over the entire wavelength range. (a) Normal incidence spectrum for the 5-layer structure. (b) Normal incidence spectrum for the 14-layer structure.

is achieved by overlapping multiple absorption resonances and with an overall graded-index structure to minimize reflection. The authors designed a 5-layer structure using MgF_2, TiO_2, Si, Ge, and Cr. The simulated average absorption of their structure over the wavelength range is 95.37% under normal incidence. If not specified otherwise, we assume normal incidence when reporting average absorption.

We hypothesize that, when choosing from a larger set of materials than used in the previous work [1], it is possible to design a structure with higher average absorption than the human-designed structure. Thus, we expanded the original material set [1] to include 11 more materials (16 total). The set of materials is listed in table 1. We set the available discrete thicknesses $\mathcal{D}$ to be $\{15, 20, 25, \ldots, 200\}$ nm with a total of 38 different values. When training the sequence generator, we set the learning rate to 5×10^{-5} and the maximum length to $L = 6$. The material embedding size d is set to 5, i.e. $emb_m \in \mathbb{R}^5$. The generator is trained for a total of 3000 epochs with the batch size set to be 1000 generation steps. We repeat the training for 10 runs with different random seeds. The best structure discovered in each run was recorded and finetuned using the quasi-Newton method.

It is worth noting that our algorithm can yield very similar structures as that reported in [1], i.e. it can search for and find the structure designed based by human experts. One of such structures is {(MgF_2, 112 nm), (TiO_2, 55 nm), (Ti, 30 nm), (Ge, 30 nm), (Cr, 200 nm)} with an average absorption of 96.12%, which has exactly the same material composition as the one reported previously [1]. However, the best structure discovered by the algorithm, exhibiting a higher average absorption of 97.64%, is {(SiO_2, 115 nm), (Fe_2O_3, 70 nm), (Ti, 15 nm), (MgF_2, 124 nm), (Ti, 148 nm)}. The spectrum under normal incidence are plotted in figure 5(a).

We plot the best absorption values before and after finetuning of all ten runs in figure 6. After finetuning, the average absorptions for the discovered structures across all runs were improved. We found that the algorithm is robust to the randomness during training as 8 out of the 10 runs achieved an absorption that is higher than 95% after finetuning.

In an additional experiment, we explore whether the algorithm can design a structure with more layers to achieve even higher absorptions. We set the maximum length $L = 15$ and sample layer materials from MgF_2, TiO_2, Si, Ge, and Cr. The best discovered structure has 14 layers with an average absorption of 99.24%. The structure configuration is summarized in table 2. We plot the normal incidence spectrum structure in figure 5(b). The structure discovered by OML-PPO reaches close-to-perfect performance under normal incidence and has high absorption over a wide range of angles.

4.2. Task 2: incandescent light bulb filter

To further test whether our method is scalable to more complicated tasks, we apply the proposed method for designing a filter that can enhance the luminous efficiency of incandescent light bulbs [40, 41]. The idea is to reflect the infrared light emitted by the light bulb filament so that its energy can be recycled. To this end, we

Table 2. RL designed 14-layer structure with 99.24% average absorption.

ID	Material	Thickness	ID	Material	Thickness
1	MgF_2	123 nm	8	Si	15 nm
2	TiO_2	32 nm	9	Cr	17 nm
3	MgF_2	21 nm	10	Ge	15 nm
4	Si	15 nm	11	TiO_2	33 nm
5	TiO_2	15 nm	12	Cr	29 nm
6	Si	15 nm	13	TiO_2	81 nm
7	Ge	15 nm	14	Cr	116 nm

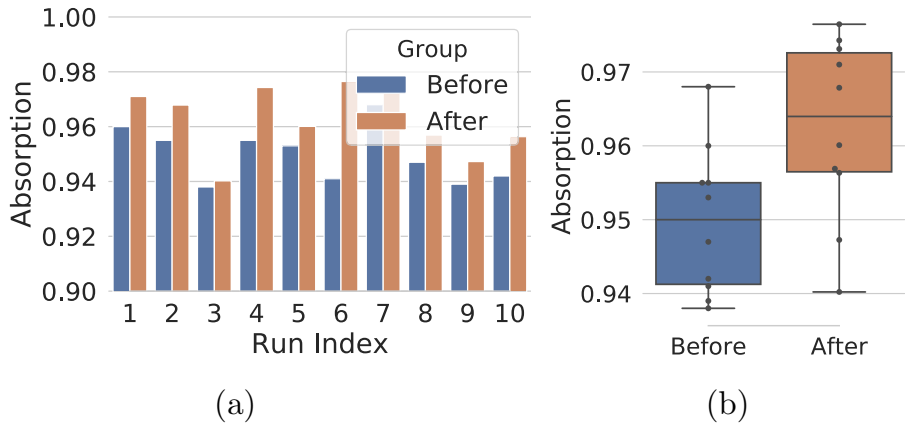

Figure 6. Absorption values before and after finetuning. finetuning improves the average absorption of every structure discovered in each run. (a) Average absorption values before and after finetuning for each individual run. (b) Box-plot for ten average absorptions values.

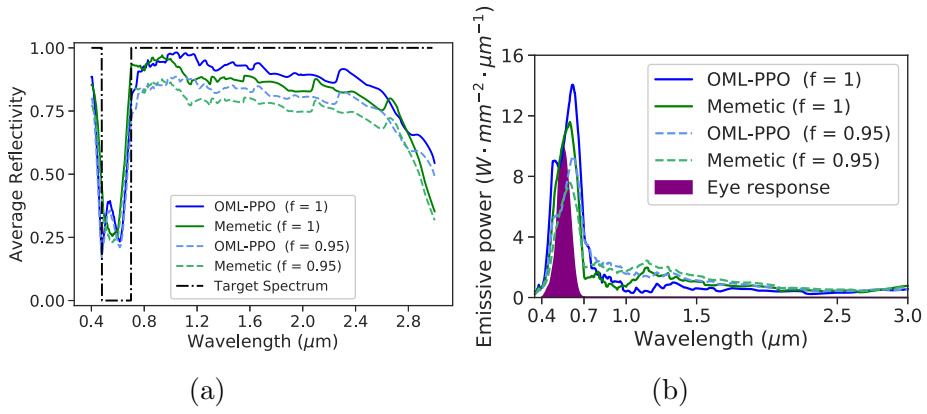

Figure 7. Results on the incandescent light bulb design. (a) Target spectrum and the average reflectivity of structures designed by OML-PPO and the memetic algorithm. (b) Emissive power spectrum. A good design will have high emissive power in the visible range (380, 780) nm. f is the view factor that equals the proportion of emitted light from the light bulb filament that can reach the light bulb filter. We report results under view factors 0.95 and 1.

set the target reflectivity to be 0% in the range (480, 700) nm, and 100% outside this range (figure 7(a)). In this way, the infrared light, which cannot contribute to lighting, will be reflected back to heat up the emitter.

A similar design has been previously studied [6, 41]. We choose the same seven dielectric materials as the available materials: Al_2O_3, HfO_2, MgF_2, SiC, SiO_2, and TiO_2 [6]. Similar to our previous experiment, we train our policy for 10 runs with different random seeds. Here, we set the maximum allowed length $L = 45$ and the learning rate to be 5×10^{-5}. The number of epochs and batch size are 10,000 and 3000, respectively. The best discovered structure is reported in table 3.

In figure 7, we compare the average reflectivity normalized over all incidence angles (0°–90°) of the 42-layer structure designed with our algorithm and the 41-layer structure designed by a memetic algorithm [6]. Our structure has a higher average reflectivity in the infrared range (>780 nm) than the 41-layer structure.

Table 3. RL designed incandescent light bulb filter with 42 layers. The total thickness is 8.54 μm.

ID	Material	Thickness	ID	Material	Thickness	ID	Material	Thickness
1	SiO_2	289 nm	15	SiC	210 nm	29	SiC	117 nm
2	SiN	268 nm	16	SiN	168 nm	30	MgF_2	224 nm
3	MgF_2	185 nm	17	MgF_2	200 nm	31	SiC	122 nm
4	SiN	189 nm	18	SiC	227 nm	32	MgF_2	235 nm
5	SiC	214 nm	19	SiN	242 nm	33	SiC	127 nm
6	SiN	214 nm	20	MgF_2	222 nm	34	MgF_2	230 nm
7	MgF_2	210 nm	21	SiC	228 nm	35	SiC	234 nm
8	SiN	206 nm	22	MgF_2	216 nm	36	MgF_2	218 nm
9	SiC	205 nm	23	SiC	229 nm	37	SiC	235 nm
10	SiN	183 nm	24	MgF_2	203 nm	38	MgF_2	220 nm
11	MgF_2	184 nm	25	SiC	101 nm	39	SiC	231 nm
12	SiN	179 nm	26	MgF_2	209 nm	40	MgF_2	216 nm
13	SiC	203 nm	27	SiC	121 nm	41	SiC	233 nm
14	SiN	273 nm	28	MgF_2	225 nm	42	Al_2O_3	95 nm

Table 4. Visible light enhancement. Our RL-designed structure achieved 8.5% higher visible light enhancement than the structure designed by a memetic algorithm.

Model	Enhancement factor
OML-PPO	16.60
Memetic [6]	15.30

Table 5. Highest absorption values discovered by each algorithm across 10 runs. The mean average absorption values and standard deviations of the 10 runs are reported.

Model	Average absorption
OML-PPO	$94.98\% \pm 0.99\%$
Only gating	94.05%±1.39%
Only auto-regressive	91.55%±1.14%
None (baseline)	91.03%±0.87%

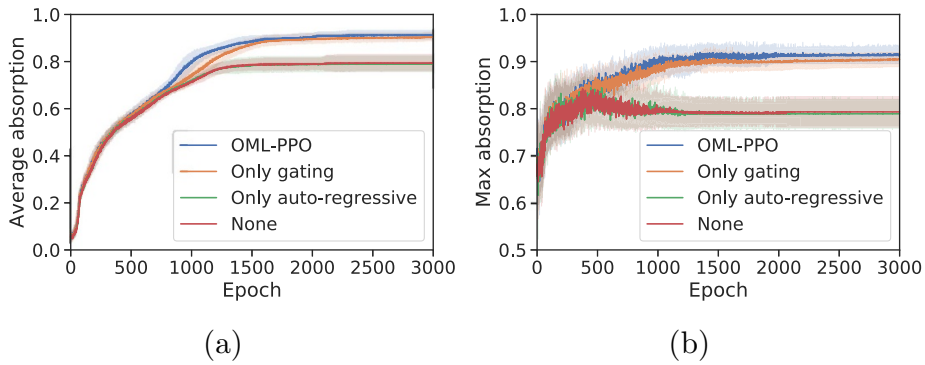

Figure 8. Training trajectory of OML-PPO and other baseline algorithms. (a) Average absorption trajectory. (b) Maximum absorption trajectory. The non-repetitive gating enables the model to converge to better solutions than models without the gating. The shaded area corresponds to one standard deviation.

We quantitatively evaluated the performance of the designed filter by calculating the enhancement factor for visible light (400–780 nm) under a fixed operating power. The results are reported in table 4. Details about the calculation of enhanced factor is included in supplementary materials.

4.3. Ablation study

On the ultra-wideband absorber design task, we conducted an ablation study to understand the effect of non-repetitive gating and auto-regressive generation of materials and thicknesses. We trained four different models: (1) OML-PPO with both non-repetitive gating and auto-regressive generation, (2) non-repetitive gating only, (3) auto-regressive generation only, and (4) neither non-repetitive gating nor the auto-regressive generation. For each model, we repeated the training for ten times. The maximum absorption values

discovered by each model before finetuning are reported in table 5. Both non-repetitive gating and the auto-regressive material/thickness generation improve the performance of the baseline model.

In figure 8, we plot the average absorption and maximum absorption of the structures generated in each epoch over the entire training trajectory. The effect of non-repetitive gating is more significant than auto-regressive material/thickness generation as the OML-PPO and the only-gating variants both significantly outperform the other two variants. The non-repetitive gating significantly improves the model convergence during training. When non-repetitive gating and the auto-regressive sampling are combined together, the model achieves the best performance.

5. Conclusion

We introduced a novel sequence generation architecture and a deep reinforcement learning pipeline to automatically design optical multi-layer films. To the best of our knowledge, our work is the first to apply deep reinforcement learning to design multi-layer optical structures with the optimal number of layers not known beforehand. Using a sequence generation network, the proposed method can select material and thickness for each layer of a multi-layer structure sequentially. On the task of designing an ultra-wideband absorber, we demonstrate that our method can achieve high performance robustly. The algorithm automatically discovered a 5-layer structure with 97.64% average absorption over the (400, 2000) nm range, which is 2% higher than a structure previously designed by human experts. When applied to generate a structure with more layers, the algorithm discovered a 14-layer structure with 99.24% average absorption, approaching perfect performance. On the task of designing incandescent light bulb filters, our method achieves 8.5% higher visible light enhancement factor than a structure designed by a state-of-art memetic algorithm. Though the spectral requirements of our two examples are simpler than some other real-life applications [42], we expect no intrinsic difficulty when applying our algorithm to tasks that require more complicated spectra. Because the reward function used in our method can be easily calculated for any arbitrarily complicated spectrum, we believe that our algorithm can be directly applied to many other multi-layer thin film design tasks with more complex spectral requirements. Moreover, with the recent development of GPUs and TPUs, reinforcement learning algorithms could become more salable than evolutionary approaches for solving complicated design tasks.

Through an ablation study, we showed that customizing the sequence generation network based on optical design domain knowledge can greatly improve the optimization performance. Our results demonstrated the high performance of the proposed method on complicated optical design tasks. Because the proposed method does not rely on hand-crafted heuristics, we believe that it can be extended to many other multi-layer optical design tasks such as lens design and multi-layer metasurface design by modifying the action space of the sequence generation network. However, for complex designs that require micro-nano structures [43], simulating the optical response can be computationally expensive. Since most deep reinforcement learning methods have a high sample complexity, it is important to develop sample-efficient reinforcement learning algorithms before such methods can be widely adopted for optical design tasks involving micro-nano structures.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/hammer-wang/oml-ppo.

Acknowledgments

H W would like to thank Rackham Graduate School at University of Michigan for funding, and Dr Yu Shi for help with calculating the enhancement factor of incandescent light bulb filters.

ORCID iD

Haozhu Wang https://orcid.org/0000-0002-9679-0144

References

[1] Yang C, Chengang J, Shen W, Lee K-T, Zhang Y, Liu X and Guo L J 2016 Compact multilayer film structures for ultrabroadband, omnidirectional and efficient absorption *ACS Photonics* **3** 590–6

[2] Agrawal M and Peumans P 2008 Broadband optical absorption enhancement through coherent light trapping in thin-film photovoltaic cells *Opt. Express* **16** 5385–96

[3] Raman A P, Anoma M A, Zhu L, Rephaeli E and Fan S 2014 Passive radiative cooling below ambient air temperature under direct sunlight *Nature* **515** 540–4
[4] Wei Li, Shi Y, Chen Z and Shanhui F 2018 Photonic thermal management of coloured objects *Nat. Commun.* **9** 1–8
[5] Schubert M F, Mont F W, Chhajed S, Poxson D J, Kim J K and Schubert E F 2008 Design of multilayer antireflection coatings made from co-sputtered and low-refractive-index materials by genetic algorithm *Opt. Express* **16** 5290–8
[6] Shi Y, Wei Li, Raman A and Fan S 2017 Optimization of multilayer optical films with a memetic algorithm and mixed integer programming *ACS Photonics* **5** 684–91
[7] You C, Matyas C T, Huang Y, Dowling J and Veronis G 2020 Optimized multilayer structures with ultrabroadband near-perfect absorption *IEEE Photonics J.* **12** 1–10
[8] Tikhonravov A V, Trubetskov M K and DeBell G W 1996 Application of the needle optimization technique to the design of optical coatings *Appl. Opt.* **35** 5493–508
[9] Rabady R I and Ababneh A 2014 Global optimal design of optical multilayer thin-film filters using particle swarm optimization *Optik* **125** 548–53
[10] Silver D *et al* 2017 Mastering the game of go without human knowledge *Nature* **550** 354–9
[11] Vinyals O *et al* 2019 Grandmaster level in Starcraft II using multi-agent reinforcement learning *Nature* **575** 350–4
[12] Bello I, Pham H, Le Q V, Norouzi M, and Bengio S 2016 Neural combinatorial optimization with reinforcement learning (arXiv:1611.09940)
[13] Khalil E, Dai H, Zhang Y, Dilkina B and Song L 2017 Learning combinatorial optimization algorithms over graphs *Advances in Neural Information Processing Systems* pp 6348–58
[14] Mirhoseini A *et al* 2017 Device placement optimization with reinforcement learning *Proc. 34th Int. Conf. Machine Learning* vol 70 pp 2430–9
[15] Mirhoseini A *et al* 2020 Chip placement with deep reinforcement learning (arXiv:2004.10746)
[16] Sutton R S and Barto A G 2018 *Reinforcement Learning: An Introduction* (Cambridge, MA: MIT Press)
[17] Hao L, Zhang X and Yang S 2020 A learning-based iterative method for solving vehicle routing problems *Int. Conf. Learning Representations*
[18] Sajedian I, Badloe T and Rho J 2019 Optimisation of colour generation from dielectric nanostructures using reinforcement learning *Opt. Express* **27** 5874–83
[19] Sajedian I, Lee H and Rho J 2019 Double-deep q-learning to increase the efficiency of metasurface holograms *Sci. Rep.* **9** 1–8
[20] Wei M, Cheng F and Liu Y 2018 Deep-learning-enabled on-demand design of chiral metamaterials *ACS Nano* **12** 6326–34
[21] Liu D, Tan Y, Khoram E and Zongfu Y 2018 Training deep neural networks for the inverse design of nanophotonic structures *ACS Photonics* **5** 1365–9
[22] Liu Z, Zhu D, Rodrigues S P, Lee K-T and Cai W 2018 Generative model for the inverse design of metasurfaces *Nano Lett.* **18** 6570–6
[23] Vinyals O, Fortunato M and Jaitly N 2015 Pointer networks *Advances in Neural Information Processing Systems* pp 2692–700
[24] Chen X and Tian Y 2019 Learning to perform local rewriting for combinatorial optimization *Advances in Neural Information Processing Systems* pp 6278–89
[25] Jiwei Li, Monroe W, Ritter A, Jurafsky D, Galley M and Gao J 2016 Deep reinforcement learning for dialogue generation *Proc. EMNLP 2016* pp 1192–202
[26] Popova M, Isayev O and Tropsha A 2018 Deep reinforcement learning for de novo drug design *Sci. Adv.* **4** eaa7885
[27] Angermueller C, Dohan D, Belanger D, Deshpande R, Murphy K and Colwell L 2020 Model-based reinforcement learning for biological sequence design *Int. Conf. Learning Representations*
[28] Jiang J, Sell D, Hoyer S, Hickey J, Yang J and Fan J A 2019 Free-form diffractive metagrating design based on generative adversarial networks *ACS Nano* **13** 8872–8
[29] Hochreiter S and Schmidhuber Jurgen 1997 Long short-term memory *Neural Comput.* **9** 1735–80
[30] Graves A 2013 Generating sequences with recurrent neural networks (arXiv:1308.0850)
[31] Chung J, Gulcehre C, KyungHyun C, and Yoshua B 2014 Empirical evaluation of gated recurrent neural networks on sequence modeling (arXiv:1412.3555)
[32] Zhu C, Byrd R H, Peihuang L and Nocedal J 1997 Algorithm 778: L-bfgs-b: fortran subroutines for large-scale bound-constrained optimization *ACM Trans. Math. Softw. (TOMS)* **23** 550–60
[33] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA: MIT Press)
[34] Byrnes S J 2016 Multilayer optical calculations (arXiv:1603.02720)
[35] Schulman J, Wolski F, Dhariwal P, Radford A, and Klimov O 2017 Proximal policy optimization algorithms (arXiv:1707.06347)
[36] Schulman J, Moritz P, Levine S, Jordan M and Abbeel P 2015 High-dimensional continuous control using generalized advantage estimation *Int. Conf. Learning Representations*
[37] Kingma D P and Jimmy B 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)
[38] Paszke A *et al* 2019 Pytorch: an imperative style, high-performance deep learning library *Advances in Neural Information Processing Systems* pp 8024–35
[39] Achiam J 2018 *Spinning Up in Deep Reinforcement Learning*
[40] Zhou J, Chen Xi and Guo L J 2016 Efficient thermal–light interconversions based on optical topological transition in the metal-dielectric multilayered metamaterials *Adv. Mater.* **28** 3017–23
[41] Ilic O, Bermel P, Chen G, Joannopoulos J D, Celanovic I and Soljačić M 2016 Tailoring high-temperature radiation and the resurrection of the incandescent source *Nat. Nanotechnol.* **11** 320
[42] Wei Li, Shi Y, Chen K, Zhu L and Fan S 2017 A comprehensive photonic approach for solar cell cooling *ACS Photonics* **4** 774–82
[43] Wei Li and Fan S 2018 Nanophotonic control of thermal radiation for energy applications *Opt. Express* **26** 15995–6021